

**REMARKS/ARGUMENTS**

Reconsideration of this application is respectfully requested.

The rejection of claims 1-23 under 35 U.S.C. §103 as allegedly made “obvious” based on Boebert et al. ‘718 in view of Bly et al. ‘657 is respectfully traversed.

As explained in applicant’s March 30, 2004 response to the first Office Action, the Examiner’s style of quoting applicant’s claim language followed by parenthetical references to text that does not appear to teach or suggest any such structure or method step is very difficult to follow and analyze. Typical questions that arise are of the kind that are indicated specifically at pages 16 and 17 of applicant’s March 30, 2004 response. However, rather than to further explain and elucidate these allegations, the new “final” Office Action for the most part merely reiterates them verbatim.

With respect to claims 1-12, for example, it would be greatly appreciated if the Examiner could more specifically identify exactly what structure, features, method steps, in Boebert allegedly teach the quoted or paraphrased claim language. Merely citing two blocks of Boebert text for each claim feature, where those blocks of text relate to several different entities in the embodiment, make it extremely difficult to understand why the Examiner considers that particular claim feature to lack novelty in light of that particular block of text.

Perhaps it would help to first provide some additional background explanation so as to help the Examiner appreciate both the cited references and the applicant’s novel improvements.

The rise of desktop computers (discussed in Boebert Column 1, line 39 to column 2, line 4) has meant that it is not practical to hold secret information in one place – instead it is often transmitted across data networks. These data networks (such as the Internet) are not secure. In

order to prevent an eavesdropper from reading the secret information as it is transmitted across a data network, it is well-known to have the sending computer encrypt the information and the recipient computer decrypt the encrypted information. Because the secret information is encrypted between the sending computer and the receiving computer, an eavesdropper eavesdropping on the communication cannot read the secret information directly.

In order to read that information, an eavesdropper would have to figure out the secret information from the encrypted information crossing the network. By far the easiest way of doing this is to carry out the same decryption process that is carried out at the receiver. However, typically that requires knowledge of a key – i.e., a secret code shared by the sender and the recipient. Keys are normally exchanged by a secure mechanism (e.g., the sender and receiver might meet up, or a secure key exchange procedure might be used).

Finding the key that was used to encrypt a message is made easier if some of that message is already known – for example, the early Enigma codes used for communication by the German Navy in the Second World War were cracked because they contained predictable text in predictable places in some exchanged messages. This is an example of the ‘known plaintext’ attack mentioned at the bottom of page 2 in the applicant’s specification.

Finding the key is often more valuable than finding an unencrypted message – the difficulty in exchanging keys securely means that they are used for a number of messages.

In order to prevent a ‘known plaintext’ attack it is necessary to ensure that an eavesdropper cannot get access to both the original message and the encrypted message.

The increasing use of insecure personal computers to generate, edit and store secret information gives an eavesdropper a further opportunity to break the encryption used to transfer

secret messages across a network. If both the original message and the encrypted message are stored in the memory of such a personal computer, then malicious code installed on that computer by the eavesdropper might be able to gain access to both the original message and the encrypted message (and send both to the eavesdropper).

The crux of the 'Bump-in-the-Wire' idea (mentioned at paragraphs beginning at page 3, line 25 of applicant's specification) is storing the original message and the encrypted message in memories accessible to different processors so that malicious code running on one of those processors cannot read both the original message and the encrypted message and thereby allow a 'known plaintext' attack to find the secret key used to encrypt the message.

However, since an unencrypted link exists between the insecure personal computer and the Bump-in-the-Wire, an eavesdropper can potentially read the original message on that link, and the encrypted message leaving the Bump-in-the-Wire by way of an insecure network. Hence, a Bump-in-the-Wire device does not prevent a 'known plaintext' attack to find the key.

The present invention overcomes this weakness of a Bump-in-the-Wire device by directly connecting it to the personal computer – thereby removing the unencrypted link between the personal computer and the Bump-in-the-Wire device. Hence, an eavesdropper must both install malicious code on the personal computer and listen to messages on the network in order to launch a 'known plaintext' attack to find the secret key used in encrypting messages from the computer to others.

The Examiner may not have yet recognized that Boebert does, in effect, disclose a Bump-in-the-Wire device in Figure 4. There, the original message leaves the insecure personal computer (workstation 40) via the communications link 48 to the trusted path subsystem 30

which encrypts the message and sends the encrypted message on to the network 50. However, the communications link 48 is insecure and hence an eavesdropper can listen to that link and the link 50 to the network in order to obtain both an original message and an encrypted message and thereby launch a 'known-plaintext' attack to find the secret key used by the processor 31 to encrypt the message.

Aside from Figure 4, Boebert works in a different way from applicant's exemplary embodiments – it encrypts data between peripherals and the workstation. That prevents an eavesdropper getting hold of the original message, but means that the processor can't work on the data (because it has been scrambled) – thus negating (when the switches 37, 38 are down) any benefits of networked computers or client-server computing. Exemplary embodiments of applicant's invention hinder a 'known plaintext' attack without taking away many of the benefits of networked computers.

With respect to claim 1, it has already been noted above that the Examiner merely repeats most of the earlier allegations verbatim without answering the questions already raised by applicant's earlier response. For example, the Examiner relies on the block of text at column 4, lines 10-42 and column 5, lines 1-9 as allegedly teaching the claimed "first interface...". However, as already previously noted, the only "interface" described in this text block is not located between the work station 40 and the network 50 – thus making Boebert '718 essentially irrelevant even with respect to the very first recitation of claim 1. In an attempt to make this irrelevance absolutely clear, claim 1 was earlier amended so as to be directed even more specifically to a "computer/network" interface device.

Apparently in an attempt to address this kind of deficiency, the Examiner now admits that Boebert “does not clearly show that there is a computer/network interface...”. However, the Examiner then drifts off into an inexplicable discussion of a computer/network interface “controlling the first and second interfaces on one display screen”. This would appear to have nothing whatever to do with any recitation anywhere in claim 1.

Nevertheless, the Examiner goes on to rely upon Bly ‘657 as allegedly teaching this alleged deficiency. The Examiner alleges that Bly uses a particular kind of user interface to control/monitor an entire network of workstations, sections, servers, etc. The Examiner goes on to allege that it would have been “obvious” to use such a user interface in the secure interface of Boebert.

It is respectfully submitted that this discussion has nothing whatever to do with the subject matter of applicant’s claim 1. Claim 1 is not directed to a user interface – it is directed to a computer/network interface that has particular utility for enhancing the security of data transmissions and receptions. Bly has absolutely nothing to do with cryptography or data security of the type here relevant. It is not clear why those of only ordinary skill in the art would find any motivation or suggestion whatsoever for combining any aspect of Bly and Boebert. Furthermore, even if they did, supplying Boebert with a user interface which controls and monitors an entire network still does nothing to teach or suggest the applicant’s claimed invention.

With respect to claim 2, the Examiner merely repeats verbatim the allegations made in the first Office Action without ever responding whatsoever to applicant’s already offered rebuttal appearing in the paragraph bridging pages 17 and 18 of the March 30, 2004 response.

With respect to claim 3, the Examiner once again quotes or paraphrases applicant's claim language followed by a parenthetical reference to text in Boebert that has no real relevance.

Where, for example, at column 5, lines 1-9 is there any transformation of received data format from a first zone at least twice prior to data processing? Even if it is assumed that Boebert workstation 40 receives encrypted packets from the trusted path subsystem and sends them to the host computer for decryption prior to display, how does this constitute transforming a data format at least twice prior to data processing?

With respect to claim 4, the Examiner once again simply reiterates verbatim allegations made in the first Office Action without even attempting to respond to applicant's rebuttal as found in the first paragraph on page 18 of the March 30, 2004 response. As noted therein, the text cited by the Examiner simply does not connect to or support the allegations apparently being made.

It is not believed necessary at this time to further encumber this response with similar specific comments directed to each of claims 5, 6, 7, 8, 9-11 and 12.

With respect to independent claim 13, the Examiner largely repeats again verbatim the erroneous assertions already made with respect to claim 1 and rebutted above. It will be noted that claim 13 has now been amended so as to be directed to a host/network interface apparatus with a first port for communication with the host using an internal data format that is used internally by the host, etc. Conforming amendments have been made in the dependent claims as well. Support for these amendments may be found, for example, at pages 7-8 et al. of the applicant's specification and in the final paragraph of the specification at pages 16 – 17 which

teach that the network interface device may be associated with something other than a computer (for example, a mobile telephone).

It is not believed necessary at this time to further comment with respect to dependent claims 14-18 since they clearly add yet further patentable distinction to the invention of their parent claim 13.

With respect to claims 19-23, the Examiner has not offered any separate grounds for rejection so it is not believed necessary to further respond at this time with respect to those claims.

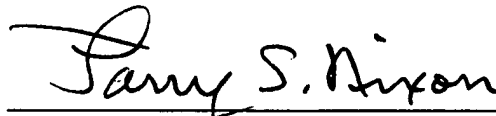
Attention is drawn to new claim 23 which will be recognized as analogous to claim 13 but now requiring the host/network interface apparatus to be adapted to be plugged into the host. This is clearly also further distinguished from any possible teaching or suggestion of the cited art.

Accordingly, this entire application is now believed to be in allowable condition and a formal Notice to that effect is respectfully solicited.

Respectfully submitted,

**NIXON & VANDERHYE P.C.**

By:

  
\_\_\_\_\_  
Larry S. Nixon  
Reg. No. 25,640

LSN:vc  
1100 North Glebe Road, 8th Floor  
Arlington, VA 22201-4714  
Telephone: (703) 816-4000  
Facsimile: (703) 816-4100